(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 359 233 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **89116956.7**

(51) Int. Cl.[5]: **G06F 13/42 , G06F 1/08**

(22) Date of filing: **13.09.89**

(30) Priority: **13.09.88 JP 229023/88**

(43) Date of publication of application:
21.03.90 Bulletin 90/12

(84) Designated Contracting States:
DE FR GB

(71) Applicant: **KABUSHIKI KAISHA TOSHIBA**
72, Horikawa-cho Saiwai-ku
Kawasaki-shi Kanagawa-ken 210(JP)

(72) Inventor: **Nakamura, Nobutaka c/o Int. Prop. Div.**
K.K. Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyio 105(JP)

(74) Representative: **Lehn, Werner, Dipl.-Ing. et al**
Hoffmann, Eitle & Partner Patentanwälte
Arabellastrasse 4
D-8000 München 81(DE)

(54) Computer system and method for changing operation speed of system bus.

(57) A computer system includes a program executing section (11, 17) for executing a program, a instruction generating section (102) for generating a frequency change instruction in response to the execution of the program, and a changing section (104) for changing the frequency of the operation clock of a system bus in response to the frequency change instruction.

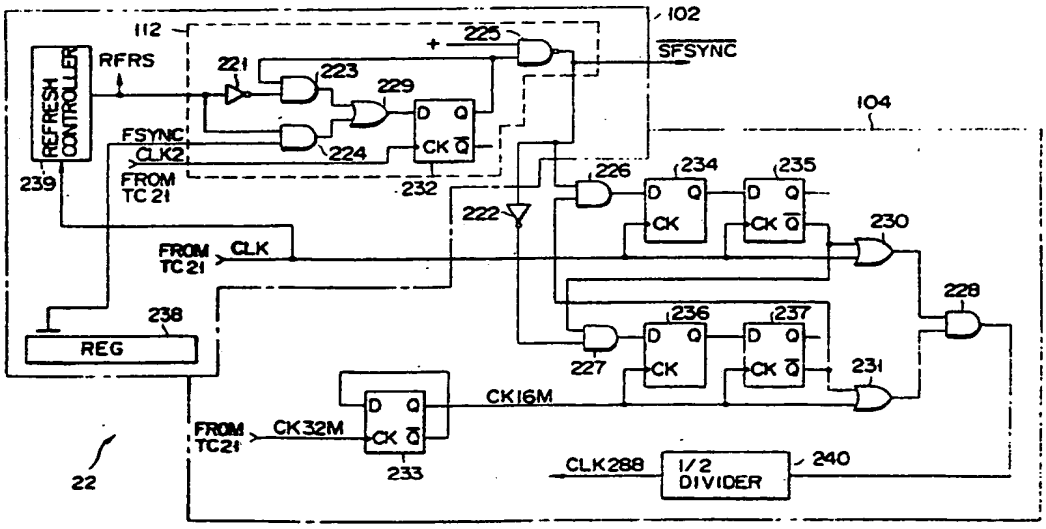

FIG. 3

EP 0 359 233 A2

Computer system and method for changing operation speed of system bus

This invention relates to a computer system and a method for changing the operation rate of a system bus.

Recently, with the development of semiconductor technology, microprocessors, memories and peripheral control LSIs can be supplied at extremely low costs. Computer systems of a relatively high performance can be constructed by adequately combining the ICs via system buses. As it has been strongly demanded to enhance the operation rate of such computer systems, the operation rate of the microprocessor which is the nucleus of the computer system is made even higher and the function thereof tends to be further enhanced.

However, the peripheral LSIs have been developed after the microprocessor was developed and it takes a longer time to develop the LSIs in general since there are many kinds of LSIs. Therefore, the operation rate of the system bus connected to the peripheral LSI has not been significantly improved. That is, in many cases, the operation rate of the system bus remains lower than that of the high-rate microprocessor so that the former cannot follow the high-rate operation of the latter. Therefore, in the conventional computer system, the frequency of the operation clock of the microprocessor tends to be set to an integer multiple of the frequency of the operation clock of the system bus, in order to easily synchronize the operation clock of the system bus with that of the microprocessor.

Further, in order to inherit the property of softwares, the newly developed computer system is required to be compatible with the conventional computer system. Some softwares are so designed with respect to hardwares which are connected to the system bus operated in response to the operation clock of a relatively low frequency. Therefore, in a case where such a hardware is used in the computer system, it is necessary to effect data transfer in a condition that the operation rate of the microprocessor is set to correspond to that of the hardware. In this case, the function of the high-speed microprocessor cannot be fully utilized.

This invention has been made in consideration of the above, and an object of this invention is to provide a computer system and a method for changing the operation rate of a system bus.

In order to attain the above object, the computer system includes a program executing section for executing a program; a instruction generating section for generating a frequency change instruction in accordance with the execution of the program; and a changing section for changing the operation clock frequency of the system bus in response to the frequency change instruction.

In order to attain the above object, the method comprises the steps of executing a program; generating a frequency change instruction in accordance with the execution of the program by a processor; and changing the operation clock frequency of the system bus in response to the frequency change instruction.

As described above, according to the computer system of this invention, the operation clock frequency of the system bus can be set in a programmable fashion.

This invention can be more fully understood from the following detailed description when taken in conjunction with the accompanying drawings, in which:

Fig. 1 is a block diagram showing the schematic construction of a computer system according to this invention;

Fig. 2 is a block diagram showing the construction of a first embodiment of the computer system according to this invention;

Fig. 3 is a block diagram showing the detail construction of that portion of a bus controller of Fig. 2 which relates to this invention;

Fig. 4 is a diagram showing the format of access control data held in a register shown in Fig. 3;

Figs. 5A to 5G are timing charts showing the operation of the first embodiment;

Fig. 6 is a block diagram showing the construction of a second embodiment of the computer system according to this invention; and

Fig. 7 is a block diagram showing the detail construction of that portion of a bus controller of Fig. 6 which relates to this invention.

There will now be described a computer system according to this invention with reference to the accompanying drawings.

In Fig. 1, a 32-bit CPU 11 is connected to a CPU bus including a CPU data bus (D31-0) 12 and a CPU address bus (A23-2) 24. The CPU 11 supplies an address to the CPU address bus 24, supplies data to the CPU data bus 12 and receives data from the CPU data bus 12. An optional arithmetic operation processor 25 is connected to the CPU data bus 12 and effects the arithmetic operation according to the instruction from the CPU 11. A B latch (B-LAT) 13 functions to connect the CPU data bus 12 to a memory data bus (MD31-0) 14 according to a control signal from the timing controller 21. The CPU data bus 12 and the CPU address bus 24 are respectively connected to a cache memory 19 and a cache memory controller 23. Data is read out from the cache memory 19 without accessing memories (DRAM) 17, 18-1 and

18-2 when a cache hit has occurred.

The memory sections (DRAM) 17, 18-1 and 18-2 are connected to the memory data bus 14 and a memory address bus (MA9-0) 26. Further, control signals RAS0 to RAS7 are supplied to the memory sections 17, 18-1 and 18-2. Each of the memory sections 17 and 18-1 has a memory capacity of 1 MB and is provided as a standard memory. The memory section 18-2 is an expansion memory and can be used as expandable memory sections of the maximum memory capacity of 12 MB. The expandable memory sections are specified in the unit of 1 MB by the signals RAS2-7 and the address thereof is designated by column and row addresses of 10 bits supplied from the timing controller 21 via the memory address bus 26. The latch (C-LAT) 15 connects the memory data bus 14 and CPU address bus 24 to a system data bus 16a and a system address bus 16b of the system bus 16 respectively in response to the timing control signal from the bus controller 22.

The timing controller 21 is connected to the CPU address bus 24, the memory address bus 26, a part of the memory data bus 14, the system address bus 16b and cache address bus (CA12-0) and generates various timing control signals to control the operation of the computer system. More specifically, the timing controller 21 generates the timing control signal for controlling the B-LAT 13 in response to control data output from the CPU 11 onto a CPU control data bus (not shown) and a transfer control signal from the bus controller 22. Further, the timing controller 21 generates a memory address and a cache memory address. The cache memory address is supplied to the cache memory 19. A bus controller 22 connected to the system bus 16 is used to perform the control on the operation on the system bus 16. More specifically, the bus controller 22 generates the timing control signal for controlling the C-LAT 15 in response to control data on a system control data bus (not shown) of the system bus 16 and a transfer control signal from the timing controller 21.

A BIOS ROM 20 stores a program for controlling the data input/output operation, i.e., a basic input/output system program. An external device 34 is connected to the system bus 16 via a serial input/output interface 32. In addition, various interfaces are connected to the system bus 16, but the explanation of those portions which do not relate to this invention is omitted.

Now, that portion of the construction of the computer system shown in Fig. 1 which relates to this invention is described in detail with reference to Fig. 2.

The memory section 17 is operated in synchronism with the input clock signal CLK and includes a memory 17-1 of 1MB and a refresh circuit 17-2 for refreshing the memory 17-1. The storage area of the memory 17-1 includes an area for storing an operating system (OS) 100 and an area for storing an application program. A system environment setting routine 102 is contained in the OS 100. The memory section 17 outputs data specified by a column address (MA9-0) and a row address (MA9-0) on the memory address bus 26 to memory buses (MD31-0) 14. The refresh circuit 17-2 generates and outputs a hold request to the CPU 11 in response to an input refresh instruction RFRS and effects the refreshing operation for the memory 17-1.

The CPU 11 is operated in synchronism with the input clock signal CLK to output an address to the CPU address bus (A23-2) 24 receive data on the memory data buses (MD31-0) 14 via the latch (B-LAT) 13 and the CPU data buses (D31-0) 12 and output data to the memory buses MD31-0. The CPU 11 executes the system environment setting routine 102 in the operating system (OS) 100 in response to a system environment setting command input upon setting up of the computer system. In the routine 102, access control data is input and held. When the computer system is started upon the normal use, the routine 102 is automatically executed and the held access control data is output the bus controller 22 via the B- and C-LATs 13 and 15. The application program in the memory 17-1 is executed by the CPU 11 under the control of the OS 100.

The B-LAT 13 latches data on the memory data bus 14 or the CPU data bus 12 in response to the timing control signal from the timing controller 21 and transfer the latched data to the CPU data buses 12 or the memory data bus 14. The C-LAT 15 latches data on the memory data bus 14 or the system data bus 16a in response to the timing control signal from the bus controller 22 and transfer the latched data to the CPU data buses 16a or the memory data bus 14. Also, at the same time, the C-LAT 15 latches an address on the CPU address bus 24 in response to the timing control signal from the bus controller 22 and transfers the latched address onto the system address bus 16b.

The timing controller 21 has a clock generator (not shown) provided therein and supplies a clock signal CLK2 of 40 MHz, a clock signal CLK of 20 MHz and a clock signal CK32M of 32 MHz to the bus controller 22. Further, the clock signal CLK is supplied to the CPU 11 and the memory section 17. The bus controller 22 receives the access control data from the CPU 11 and changes the operation clock frequency of the system bus 16 according to the frequency change bit in the access control data while the refresh operation is effected. The controller 22 also generates and outputs the refresh instruction signal RFRS to the

memory section 17.

The detail construction of the bus controller 22 is explained with reference to Fig. 3.

The bus controller 22 includes a refresh controller 239, an access control data register 238, inverters 221 and 222, AND gates 223 to 228, OR gates 229 to 231, D-type flip-flops (F F) 232 to 237 and a 1 2 frequency divider 240. The inverter 221, AND gates 223, 224 and 225 and F·F 232 constitutes a frequency change instruction generating section 112, and the refresh controller 239, the register 238 and the frequency change instruction generating section 112 constitutes a instruction generator section 102. Further, the inverter 222, the AND gates 226 to 228, the F Fs 234 to 237, the OR gates 230 and 231 and the frequency divider 240 constitutes a frequency change section.

The refresh controller 239 generates a refresh instruction signal RFRS at a regular interval of time in response to the signal CLK from the timing controller 21 and supplies the refresh instruction signal RFRS to the inverter 221, the AND gate 224 and the refresh circuit 17-2 of the memory section 17. The register 238 receives access control data from the CPU 11 via the system bus 16 and holds the same therein. The format of the access control data is shown in Fig. 4. A sixth bit of the access control data is a data bit FSYNC which relates to this invention. The bit is set to a logic "0" when the system bus is to be operated at a frequency of 10 MHz and to a logic "1" when the system bus is to be operated at a frequency of 8 MHz. The other data bits have no relation to this invention and the explanation thereof is omitted. The bit data FSYNC is supplied to the AND gate 224.

The AND gate 223 receives an output of the inverter 221 and a Q output of the F·F 232 and supplies an output corresponding to the result of the logical AND operation to the OR gate 229. The AND gate 224 supplies an output corresponding to the logical product of the data bit SFSYNC held in the register 238 and the signal RFRS to the OR gate 229. An output of the OR gate 229 is supplied to the D input terminal of the F·F 232. The clock input terminal of the F·F 232 is supplied with the clock signal CLK2 of 40 MHz from the timing controller 21. A Q output of the F·F 232 is supplied to the AND gates 223 and 225. The other input terminal of the AND gate 225 is connected to a positive power source terminal.

The clock signal CLK of 20 MHz is supplied to the clock input terminals of the F·Fs 234 and 235 and to the OR gate 230. The signal $\overline{\text{SFSYNC}}$ is supplied to the AND gate 226 and the inverter 222. The AND gate 226 is further supplied with a $\overline{Q}$ output of the F F 237. The AND gate 226 supplies an output corresponding to the logical product of the signal $\overline{\text{SFSYNC}}$ and the $\overline{Q}$ output of the F F 237 to the D input terminal of the F F 234. A Q output of the F F 234 is supplied to the D input terminal of the F·F 235. A $\overline{Q}$ output of the F F 235 is supplied to the OR gate 230 and AND gate 227. The OR gate 230 supplies an output corresponding to the logical sum of the $\overline{Q}$ output of the F F 235 and the signal CLK to the AND gate 228.

The frequency of the clock signal CK32M of 32 MHz is divided into 16 MHz by the F F 233 and then the clock signal is supplied to the clock input terminals of the F·Fs 236 and 237 and to the OR gate 231. An output of the inverter 222 is supplied to the AND gate 227. As described before, the AND gate 227 is supplied with the $\overline{Q}$ output of the F·F 235. The AND gate 227 supplies an output corresponding to the logical product of an inverted signal of the signal $\overline{\text{SFSYNC}}$ and the $\overline{Q}$ output of the F·F 235 to the D input terminal of the F F 236 whose Q output is in turn supplied to the D input terminal of the F F 237. A $\overline{Q}$ output of the F F 237 is supplied to the OR gate 231 and AND gate 226. The OR gate 231 supplies an output corresponding to the logical sum of the $\overline{Q}$ output of the F F 237 and the signal CK16M to the AND gate 228.

The AND gate 228 supplies an output corresponding to the logical product of the outputs of the OR gates 230 and 231 to the 1·2 frequency divider 240. The frequency divider 240 divides the frequency of an output signal of the AND gate 228 by 2 and supplies the (1/2)-divided frequency signal as a signal CLK288 to the system bus 16.

Now, the operation of the first embodiment of this invention is explained with reference to Figs. 5A to 5G.

First, the power source is turned on to start up the computer system and the operating system (OS) 100 is started. When the system environment setting routine 102 in the OS 100 is executed, the access control data held therein is read out and is supplied from the CPU 11 to the bus controller 22 via the B-LAT 13 and C-LAT 15. As a result, the access control data is held in the register 238. The access control data is set and held in the system environment setting routine 102 when the computer system is set up and the routine 102 is executed in response to a system environment setting command input from a keyboard. When the system bus 16 is operated at a frequency of 10 MHz, the sixth data bit FSYNC of the access control data is set at the logic level "0". When the system bus 16 is operated at a frequency of 8 MHz, the sixth data bit FSYNC is set at the logic level "1".

When the bit FSYNC is at the logic level "0", the Q output of F·F 232 is set to the logic level "0" and the signal $\overline{\text{SFSYNC}}$ is set to the logic level "1". When the signal SFSYNC is set to the logic level "1", the $\overline{Q}$ output of the F F 237 is set to the logic level "1" after one clock of the clock signal CK16M

i.e. at a timing when the second clock is rising. Therefore, after the one clock of the clock signal CLK, the $\overline{Q}$ output of the F/F 235 is set to the logic "0". As a result, the clock signal CLK is supplied from the OR gate 230 to the AND gate 228. At this time, since an output of the OR gate 231 is set at the logic level "1", an output of the AND gate 228 becomes a clock signal of 20 MHz which is in turn divided by means of the frequency divider 240 to produce a clock signal of 10 MHz as a signal CLK288.

In a case where the operation clock of the system bus 16 is to be set to 8 MHz in the system environment setting routine 102, the access control data includes the bit FSYNC of logic "1". The access control data is supplied from the CPU 11 and set into the register 238. At this time, even if the Q output of the F/F 232 is set at the logic "0", the Q output of the F/F 232 is always set to the logic level "1" if an instruction signal RFRS of logic "1" is generated from the refresh controller 239 as shown in Fig. 5C. Then, the Q output of the F/F 232 is kept set at the logic "1". As a result, the signal $\overline{SFSYNC}$ is set to the logic "0" as shown in Fig. 5E.

Even if the $\overline{Q}$ output of the F/F 235 is set at the logic level "0", when the signal $\overline{SFSYNC}$ is set to the logic level "0", the $\overline{Q}$ output of the F/F 235 is set to the logic level "1" after one clock of the clock signal CLK since the $\overline{Q}$ output of the F/F 237 is set at the logic "1". At this time, since the $\overline{Q}$ output of the F/F 237 is kept set at the logic "1", two input signals of the logic level "1" are supplied to the AND gate 228. As a result, as shown in Fig. 5F, an output of the AND gate is kept set at the logic "1". Further, when the signal SFSYNC is set at the logic "0", the $\overline{Q}$ output of the F/F 237 is set to the logic level "0" after one clock of the signal CK16M when the $\overline{Q}$ output of the F/F 235 has been set to the logic level "1". Therefore, the OR gate 231 permits the clock signal CK16M to pass therethrough. As a result, as shown in Fig. 5F, the clock signal appears again. The clock signal is the signal CK16M and is divided by means of the frequency divider 240 and supplied to the system bus 16 as a clock signal of 8 MHz.

As described above, in the first embodiment, the frequency of the operation clock of the system bus can be programmatically set to the frequency which is set in the system environment setting routine at the time of starting up the system.

Now, the construction of the second embodiment of this invention is explained with reference to Fig. 6.

Since the second embodiment is similar to the first embodiment, only that portion of the second embodiment which is different from the first embodiment is explained. Assume that a CPU 11 transfers data to an external device 34 via an interface 32 which can be operated only at a low clock frequency while the application program is executed. When the CPU 11 executes an output command in the application program for the interface 32, the CPU 11 generates and supplies an output instruction to a DMAC 30 according to a basic input/output system program stored in the BIOS ROM 20. The DMAC 30 reads out data from a memory section 17 in response to the output instruction and transfers the read-out data to the external device 34 via the interface 32. When all the data is completely transferred, the DMAC 30 generates a completion notice to the CPU 11. A bus controller 22' monitors the output instruction from the CPU 11 to the DMAC 30 and the completion notice from the DMAC 30 to the CPU 11, and programmatically sets the operation clock frequency of the system bus from 10 MHz to 8 MHz. When detecting the output instruction to the interface 32 the bus controller 22' changes the operation clock frequency of the system bus 16 to 8 MHz and maintains the frequency of 8 MHz until the completion notice is input from the DMAC 30. While the the operation clock frequency is changed, there is a possibility that a next command will be executed since the CPU 11 executes commands of the application program in a pipeline fashion. At this time, in order to prevent the next command from being executed, a WAIT signal is supplied from the bus controller 22' to the CPU 11 so as to set the CPU 11 into the wait state.

Next, the construction of the bus controller 22' is explained with reference to Fig. 7. As shown in Fig. 7, the bus controller 22' includes an address comparator 250, a decoder 252, a J-K flip-flop (F/F) 254 and an AND gate 225 in order to generate a signal $\overline{SFSYNC}$. The address comparator 250, the decoder 252, the F/F 254 and the AND gate 225 constitute a instruction generator section 106.

The address comparator 250 receives an address on the system address bus 16b and comparing the received address with held addresses to determined whether the received address coincides one of the held addresses. The held address is predetermined in the system environment setting routine 102 when the computer system is set up and is supplied to the comparator 250 upon start of the computer system. When a coincidence is obtained, e.g., when it is detected that the received address is coincided with an address of the DMAC 30, the address comparator 250 outputs a control signal to the decoder 252. The decoder 252 receives and decodes data or the output instruction on the system data bus 16a in response to the control signal from the comparator 250 and checks whether it is an input instruction or output instruction.

When the received data is the output instruction to the DMAC 30, the decoder 252 supplies a signal of the logic "1" to the J-terminal of the F F 254. As a result, the Q output of the F F 254 is set to the logic "1" and at the same time the signal $\overline{SFSYNC}$ is set to the logic "0".

Then, the same operation is effected as in the first embodiment, and the operation clock of the system bus 16 is changed to 8 MHz. A WAIT signal generator 256 generates to the CPU 11 a signal WAIT for inhibiting the operation of the CPU 11 for a predetermined period of time in response to the signal $\overline{SFSYNC}$.

Thereafter, all the data has been transferred, the DMAC 30 supplies the completion notice to the CPU 11. The completion notice is also supplied to the K terminal of the F F 254 and the F F 254 is reset, thus setting the Q output of the F F 254 to the logic "0". As a result, the system bus 16 is operated at the frequency of 10 MHz again. The timings in the above explanation are the same as those shown in Figs. 5A to 5G.

As described above, in the second embodiment, since the frequency of the operation clock of the system bus is automatically changed to a lower frequency only when the interface having a low operation rate is accessed while the application program is executed. Therefore the average operation speed of the system can be maintained high.

It is possible to specify the I/O interface which necessitates change of the frequency of the operation clock of the system bus in the system environment setting routine and set the address thereof into the address comparator. And, in the above embodiment, only the output instruction is described but the input instruction is similar.

Reference signs in the claims are intended for better understanding and shall not limit the scope.

Claims

1. A computer system in which the operation clock frequency of a system bus can be changed characterized by comprising:
program executing means (11, 17) for executing a program;
instruction generating means (102) for generating a frequency change instruction in response to the execution of the program; and
changing means (104) for changing the operation clock frequency of the system bus in response to the frequency change instruction.

2. A system according to claim 1, characterized in that said program is for setting the system environment and is part of the operating system.

3. A system according to claim 1, characterized in that said program executing means (11, 17) includes memory means (17) for storing the program; and
said instruction generating means (102) includes:
register means (238) for holding change control data;
refresh instruction generating means (239) for generating a refresh instruction for specifying a refresh process of said memory means (17); and
frequency change instruction generating means (112) for generating the frequency change instruction in response to the change control data and the refresh instruction.

4. A system according to claim 3, characterized in that said program executing means includes:
memory means (17, 20) for storing a program; and
a CPU (11) for executing the program; said CPU (11) including means for setting the change control data into said register means (238) in response to the execution of the program.

5. A system according to claim 1, characterized in that said program is an application program and includes an input/output command to an input/output device.

6. A system according to claim 5, characterized in that said instruction generating means (106) includes:
means (250, 252) for generating a setting instruction in response to the input/output instruction; and
frequency changing instruction generating means (254) for generating the frequency change instruction in response to the setting instruction and input reset instruction; said input/output device generates the reset instruction at the time of completion of the input/output process corresponding to the input/output command.

7. A system according to claim 6, characterized in that said program executing means includes:
memory means (17, 20) for storing the program; and
a CPU (11) for executing the program; said CPU (11) including means for generating an input/output instruction in response to the execution of the program.

8. A system according to claim 7, characterized in that said input/output device is determined by a routine for setting the system environment, the routine being part of an operating system.

9. A system according to claim 7, characterized by further comprising wait signal generating means (256) for generating and outputting a wait signal to said CPU (11) in response to the frequency change instruction, said processor (11) interrupting the execution of the program in response to the wait signal.

10. A system according to claim 1, characterized in that said changing means (104) further

comprises means (234, 235, 236, 237, 230, 231) for inhibiting the output of at least one clock before the operation clock frequency is changed.

11. A method for changing the operation clock frequency of a system bus characterized by comprising the steps of:
executing a program:
generating a frequency change instruction in response to the execution of the program by a processor; and
changing the operation clock frequency of the system bus in response to the frequency change instruction.

12. A method according to claim 11, characterized in that said program is for setting the system environment and is part of the operating system.

13. A method according to claim 12, characterized in that said step of generating the frequency change instruction includes the steps of:
generating a refresh instruction for specifying the refresh process for memory means; and
generating a frequency change instruction in response to the change control data and refresh instruction.

14. A method according to claim 13, characterized in that said step of executing the program includes the steps of:
and executing the program stored in memory means; and
generating change control data according to the execution of the program.

15. A method according to claim 11, characterized in that said program is an application program and includes an input/output instruction with respect to an input/output device.

16. A method according to claim 15, characterized in that said step of generating the frequency change instruction includes the steps of:
generating a setting instruction in response to the input/output instruction;
generating a reset instruction at the time of completion of the input/output process; and
generating the frequency change instruction in response to the setting instruction and reset instruction.

17. A method according to claim 16, characterized in that said step of executing the program includes the steps of:
executing the program; and
generating an input/output instruction according to the execution of the program.

18. A method according to claim 17, characterized in that said input/output device is determined by a routine for setting the system environment, the routine being part of the operating system.

19. A method according to claim 17, characterized in that said step of generating the frequency change instruction includes a step of generating a wait signal in response to the frequency change instruction, and said program execution step includes a step of interrupting the execution of the program in response to the wait signal.

20. A method according to claim 11, characterized by further comprising a step of inhibiting the output of clock immediately before the operation clock frequency is changed.

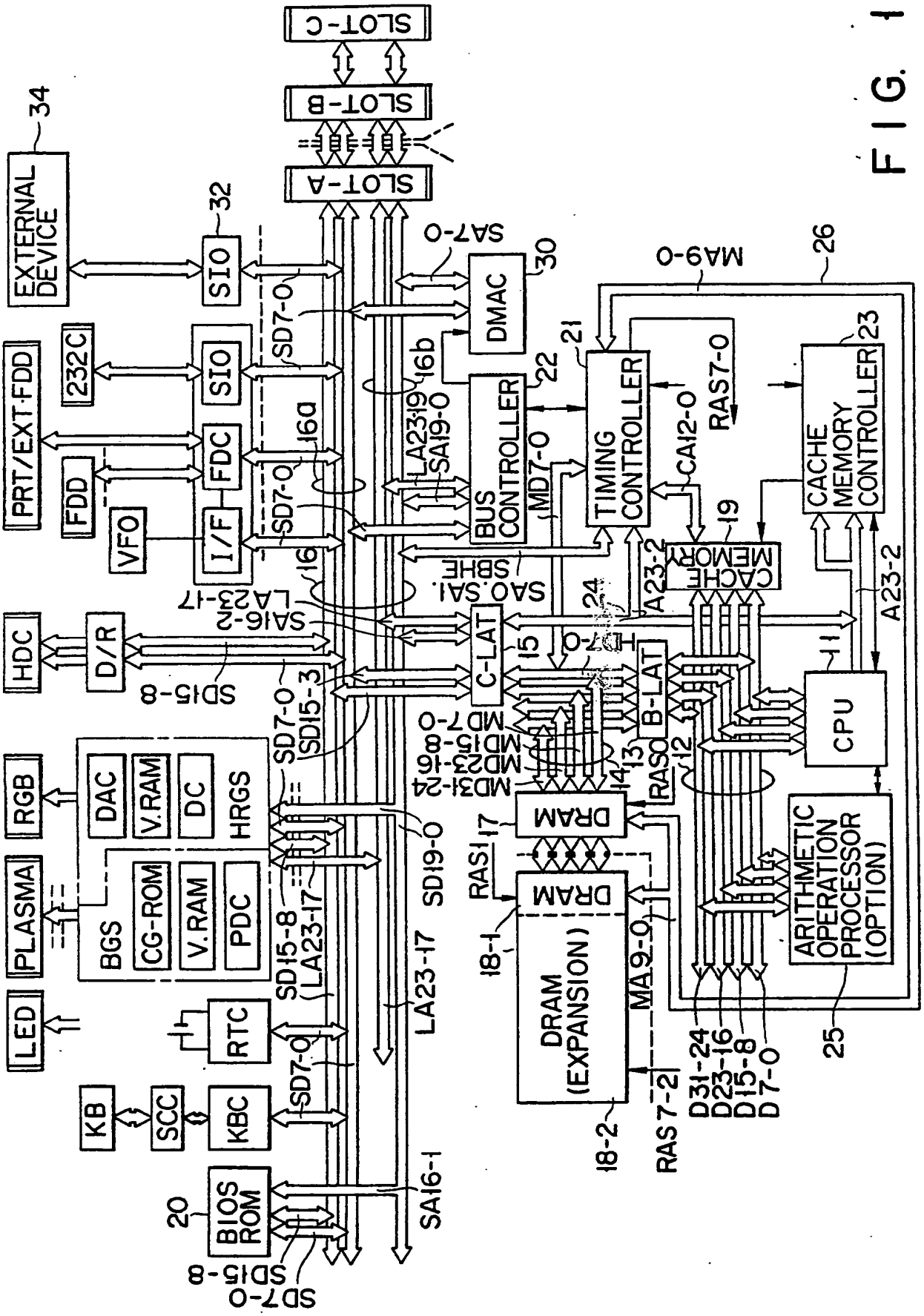
SLOT-C
SLOT-B
SLOT-A
EXTERNAL DEVICE
34
SIO
32
SA7-0
DMAC
30
MA9-0
26
SD7-0
21
22
TIMING CONTROLLER
CACHE MEMORY CONTROLLER
23
232C
PRT/EXT·FDD
SIO
16b
LA23-19
SA19-0
BUS CONTROLLER
MD7-0
RAS7-0
CA12-0
FDD
FDC
16a
VFO
I/F
SD7-0
SBHE
SA0
CACHE MEMORY
19
A23-2
16
LA23-17
SA16-2
24
HDC
D/R
SD15-8
C-LAT
15
MD7-0
B-LAT
11
CPU
SD7-0
SD15-3
MD15-8
MD23-16
MD31-24
13
14
12
RAS0
RGB
DAC
V.RAM
DC
HRGS
DRAM
17
RAS1
PLASMA
BGS
CG-ROM
V.RAM
PDC
SD15-8
LA23-17
SD19-0
DRAM (EXPANSION)
18-1
MA9-0
ARITHMETIC OPERATION PROCESSOR (OPTION)
LED
RTC
SD7-0
LA23-17
25
D31-24
D23-16
D15-8
D7-0
KB
SCC
KBC
SD7-0
18-2
RAS7-2
SA16-1
20
BIOS ROM
SD15-8
SD7-0

FIG. 1

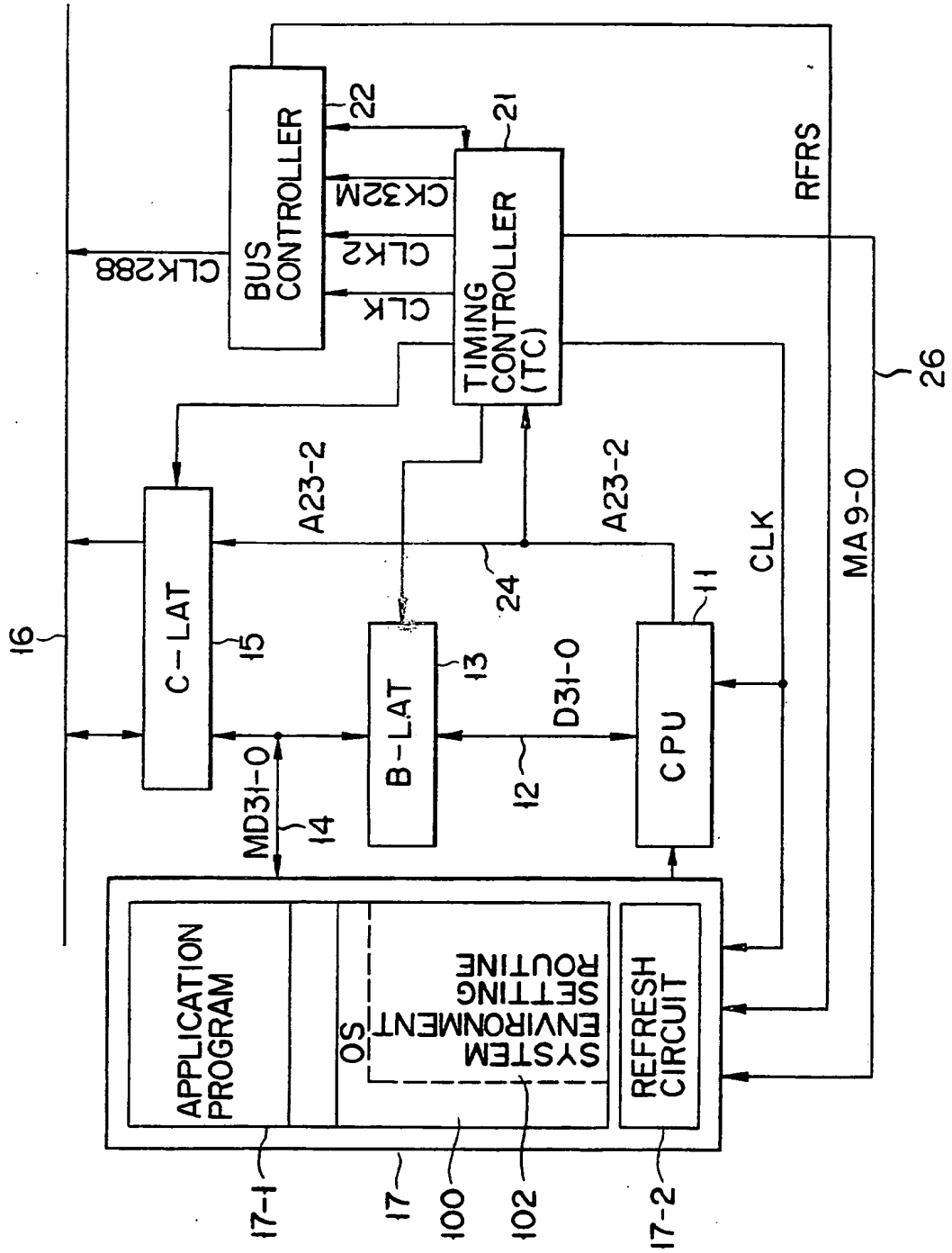
BUS CONTROLLER
TIMING CONTROLLER (TC)
CLK288
CK32M
CLK2
CLK
RFRS
MA9-0
26
22
21
A23-2
A23-2
24
C-LAT
B-LAT
CPU
CLK
11
16
15
13
D31-0
12
MD31-0
14
APPLICATION PROGRAM
OS
SYSTEM ENVIRONMENT SETTING ROUTINE
REFRESH CIRCUIT
17-1
17
100
102
17-2

FIG. 2

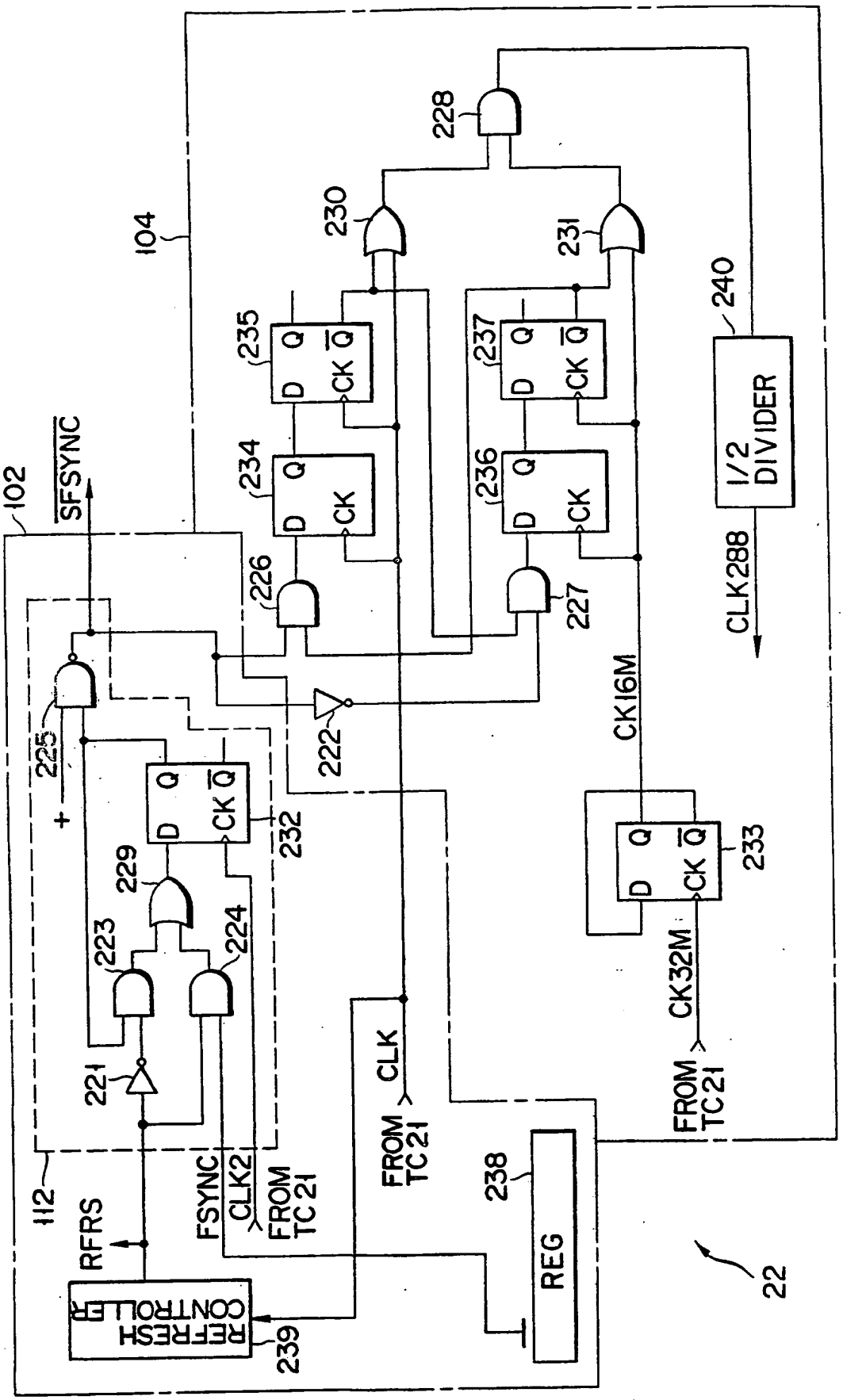
104
102
SFSYNC
112
RFRS
FSYNC
CLK2
FROM TC21
CLK
FROM TC21
CK16M
CK32M
FROM TC21
CLK288
1/2 DIVIDER
REFRESH CONTROLLER
REG
22
225
222
232
229
223
224
221
226
227
228
230
231
235
234
237
236
240
238
239
233

FIG. 3

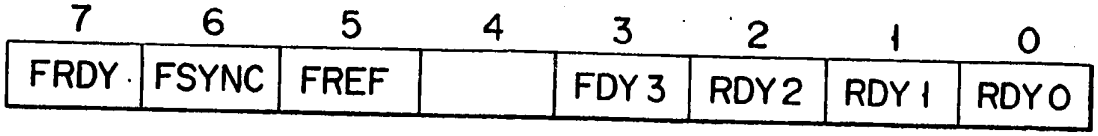

FIG. 4

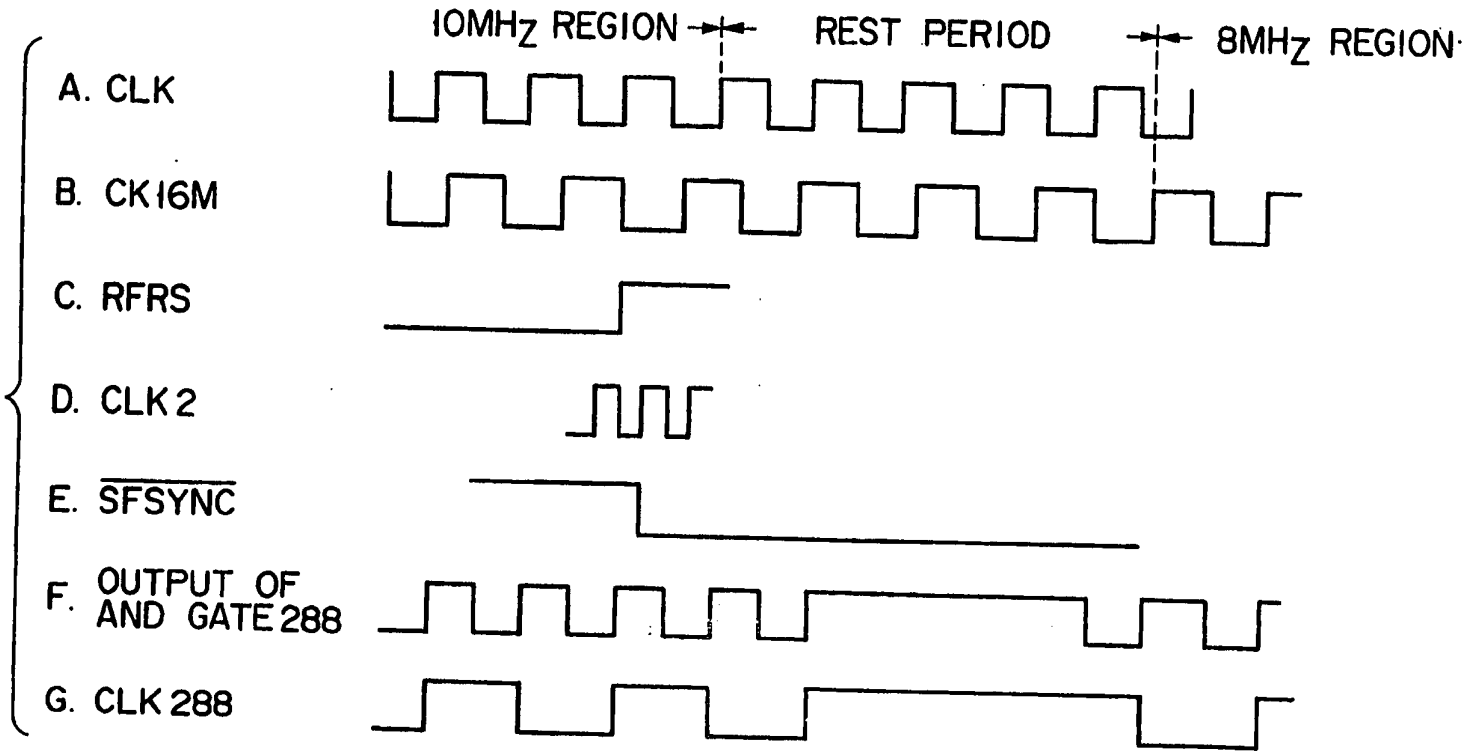

FIG. 5

EP 0 359 233 A2

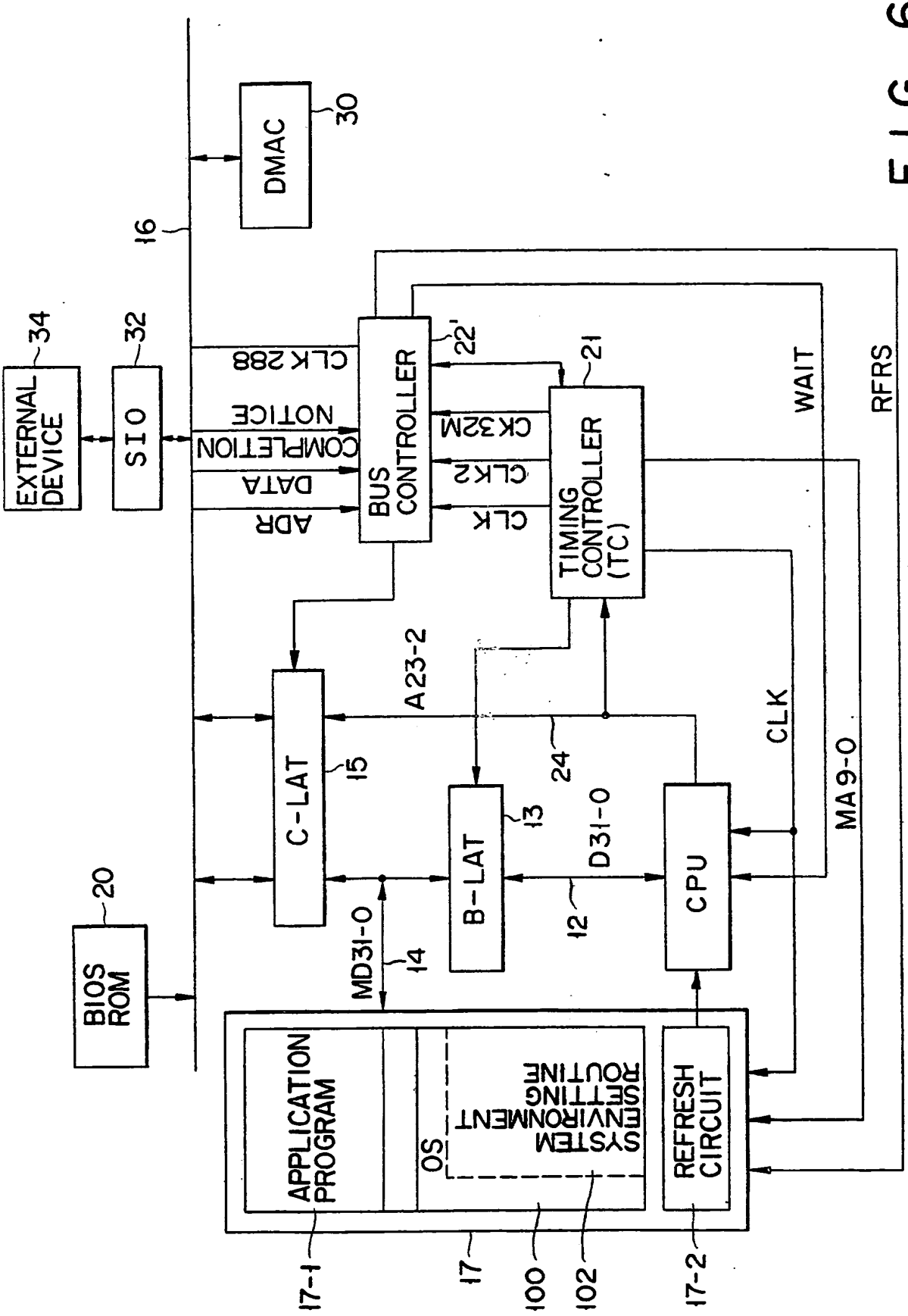
DMAC
30
16
EXTERNAL DEVICE
34
SIO
32
CLK 288
NOTICE
COMPLETION
DATA
ADR
BUS CONTROLLER
22'
CK32M
CLK2
CLK
TIMING CONTROLLER (TC)
21
WAIT
RFRS
A23-2
C-LAT
15
24
B-LAT
13
D31-0
CPU
12
CLK
MA9-0
MD31-0
14
BIOS ROM
20
APPLICATION PROGRAM
OS
SYSTEM ENVIRONMENT SETTING ROUTINE
REFRESH CIRCUIT
17-1
17
100
102
17-2

FIG. 6

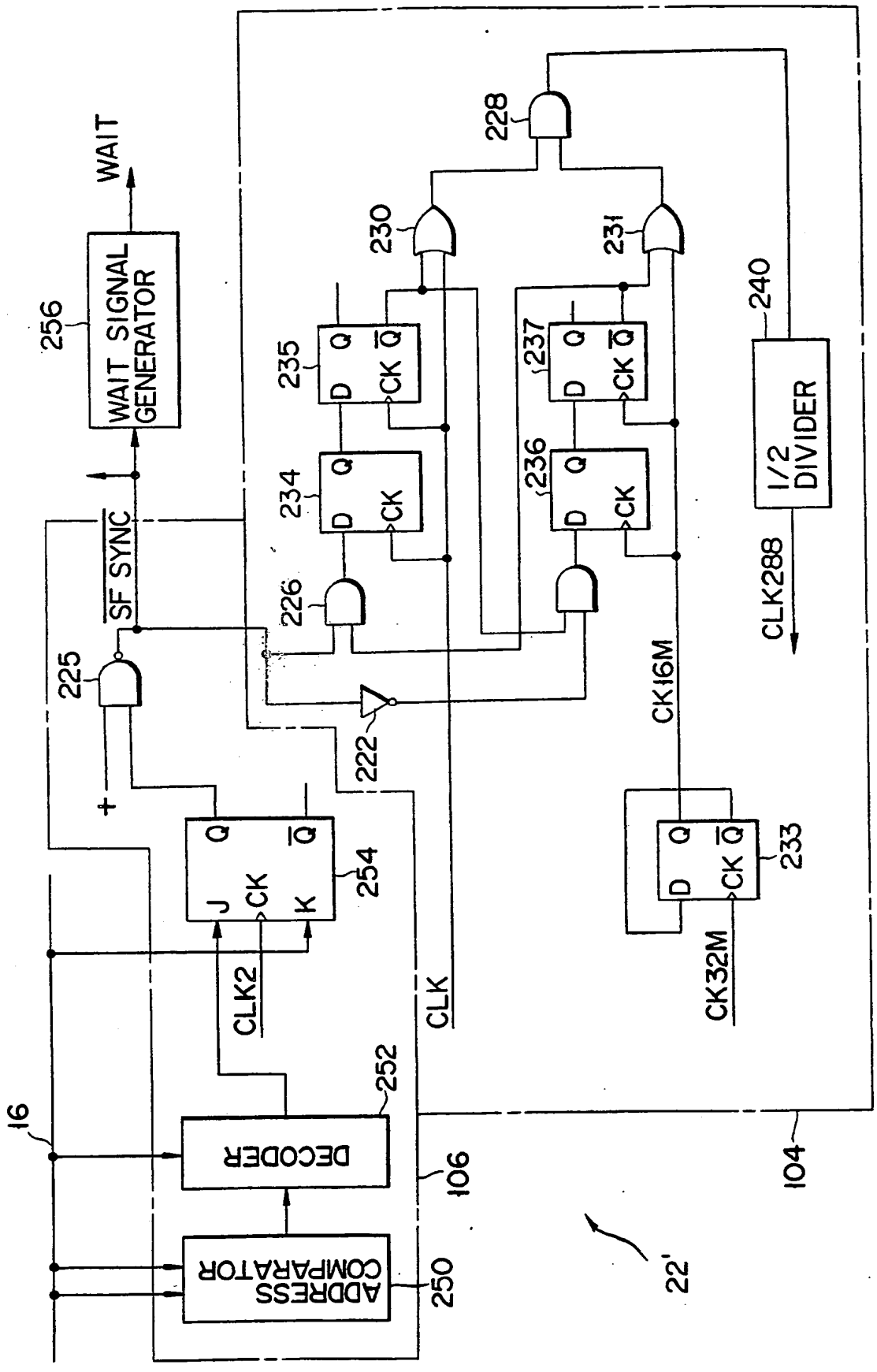
WAIT
WAIT SIGNAL GENERATOR
256
SF SYNC
225
226
228
230
231
234
235
236
237
240
1/2 DIVIDER
CLK288
CK16M
222
254
233
CLK2
CLK
CK32M
252
DECODER
16
106
104
ADDRESS COMPARATOR
250
22'

FIG. 7

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 359 233 A3**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 89116956.7

(51) Int. Cl.5: G06F 13/42, G06F 1/08

(22) Date of filing: 13.09.89

(30) Priority: 13.09.88 JP 229023/88

(43) Date of publication of application:
21.03.90 Bulletin 90/12

(84) Designated Contracting States:
DE FR GB

(88) Date of deferred publication of the search report:
17.07.91 Bulletin 91/29

(71) Applicant: KABUSHIKI KAISHA TOSHIBA
72, Horikawa-cho Saiwai-ku
Kawasaki-shi Kanagawa-ken 210(JP)

(72) Inventor: Nakamura, Nobutaka c/o Int. Prop. Div.
K.K. Toshiba 1-1 Shibaura 1-chome
Minato-ku Tokyo 105(JP)

(74) Representative: Lehn, Werner, Dipl.-Ing. et al
Hoffmann, Eitle & Partner Patentanwälte
Arabellastrasse 4
W-8000 München 81(DE)

(54) Computer system and method for changing operation speed of system bus.

(57) A computer system includes a program executing section (11, 17) for executing a program, a instruction generating section (102) for generating a frequency change instruction in response to the execution of the program, and a changing section (104) for changing the frequency of the operation clock of a system bus in response to the frequency change instruction.

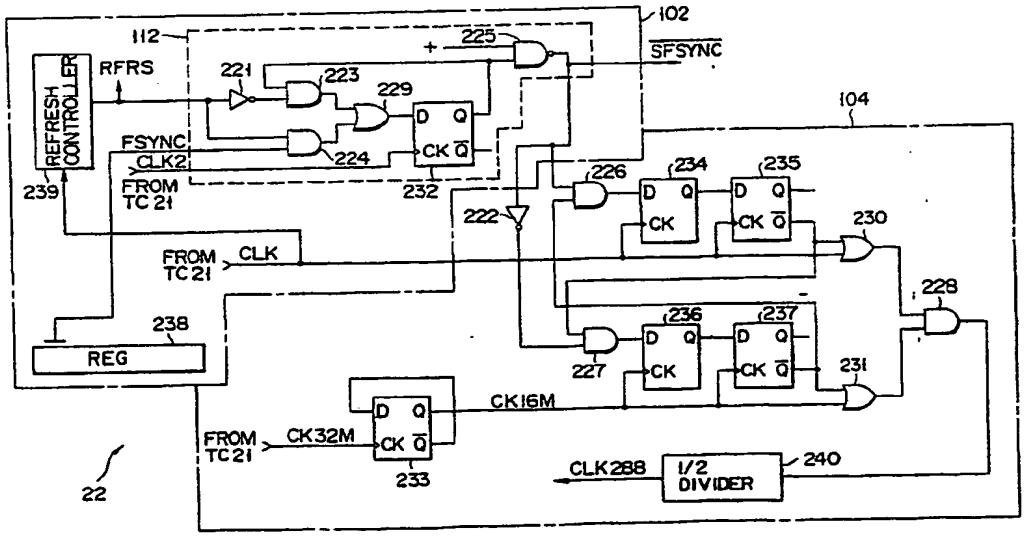

FIG. 3

EP 0 359 233 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 89 11 6956

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
X	PATENT ABSTRACTS OF JAPAN, vol. 12, no. 175 (P-707)[3022], 25th May 1988; & JP-A-62 286 117 (NEC CORP.) 12-12-1987 * Abstract *	1-2,11-12	G 06 F 13/42 G 06 F 1/08
A	PATENT ABSTRACTS OF JAPAN, vol. 12, no. 458 (P-794)[3305], 2nd December 1988; & JP-A-63 181 018 (MATSUSHITA ELECTRIC) 26-07-1988 * Abstract *	1,11	
A	PATENT ABSTRACTS OF JAPAN, vol. 9, no. 127 (P-360)[1850], 31st May 1985; & JP-A-60 10 318 (SANYO DENKI) 19-01-1985 * Abstract *	1,11	
A	EP-A-0 097 977 (DATA GENERAL CORP.) * Abstract; page 1, line 32 - page 2, line 29; figure 1 *	1,3,11,13	
A	GB-A-2 192 295 (LOGICA) * Page 1, lines 48-59; page 7, line 91 - page 8, line 24; figures 1-3 *	1,11	

TECHNICAL FIELDS SEARCHED (Int. Cl.5)

G 06 F 13
G 06 F 1

The present search report has been drawn up for all claims

Place of search	Date of completion of search	Examiner
The Hague	19 April 91	MCDONAGH F.M.

CATEGORY OF CITED DOCUMENTS

X: particularly relevant if taken alone
Y: particularly relevant if combined with another document of the same catagory
A: technological background
O: non-written disclosure
P: intermediate document
T: theory or principle underlying the invention

E: earlier patent document, but published on, or after the filing date
D: document cited in the application
L: document cited for other reasons

&: member of the same patent family, corresponding document